

FIG. 1

200

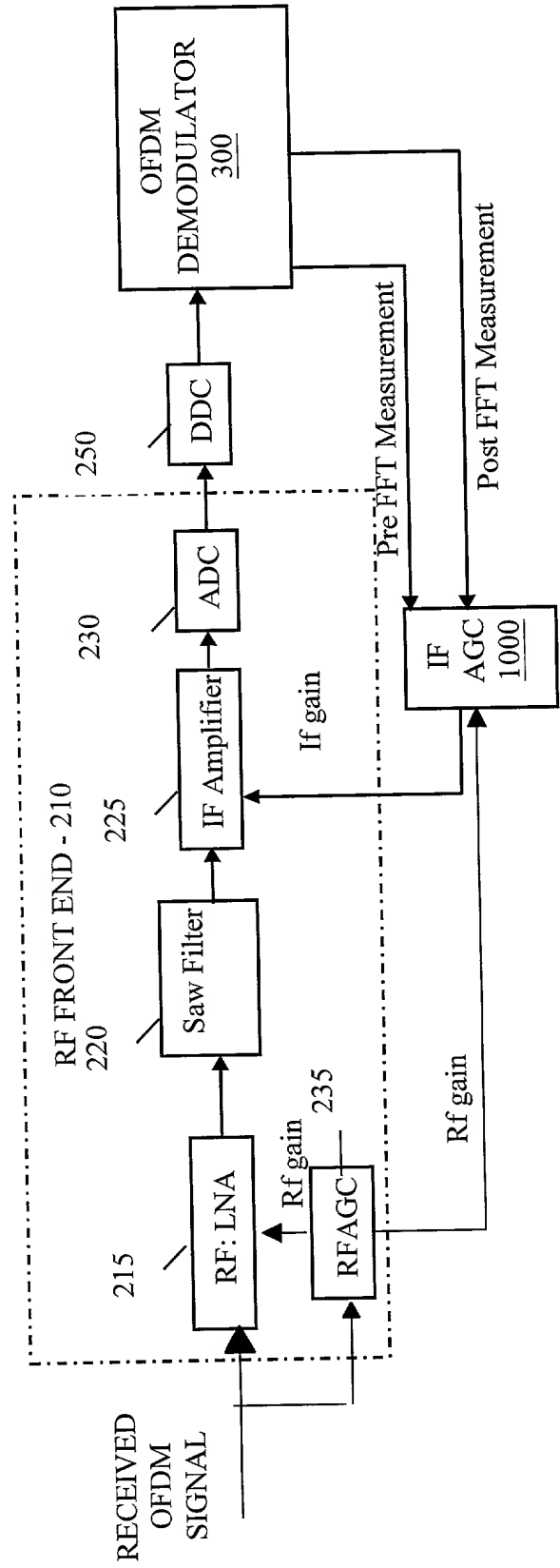


FIG. 2

PRE-FFT MEASUREMENT

POST-FFT MEASUREMENT

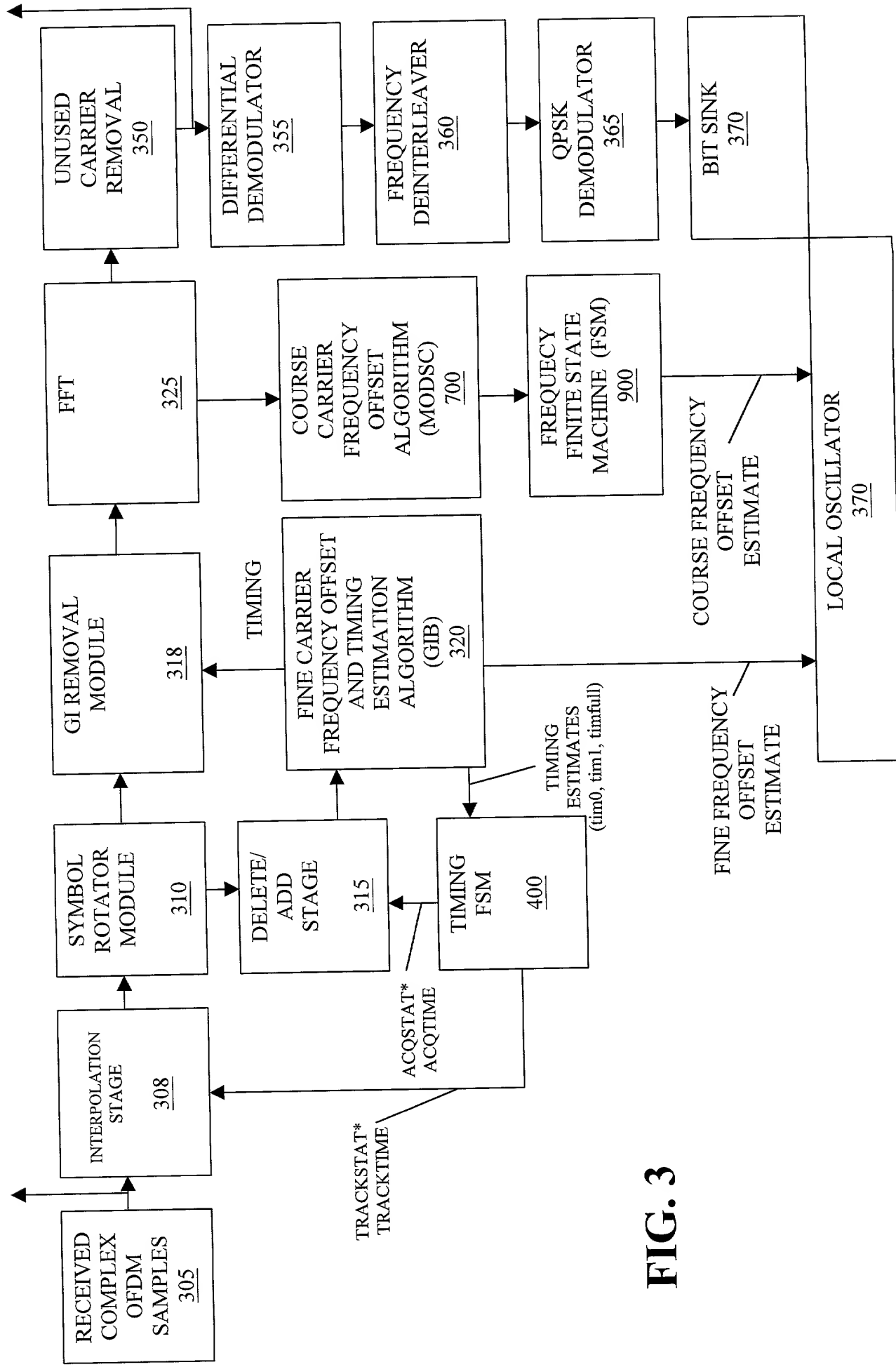


FIG. 3

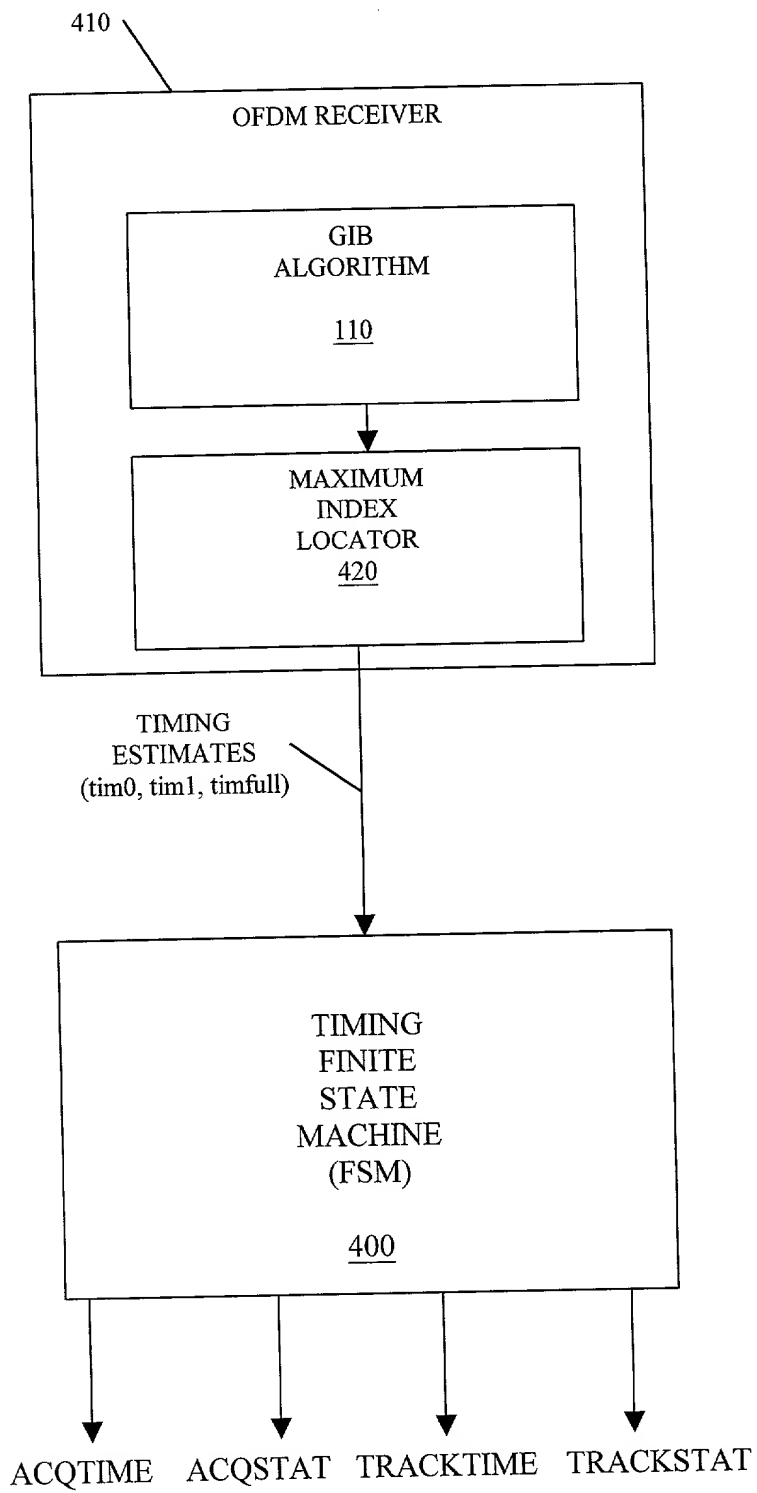


FIG. 4

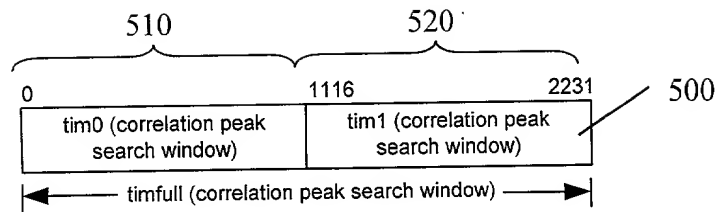


FIG. 5A

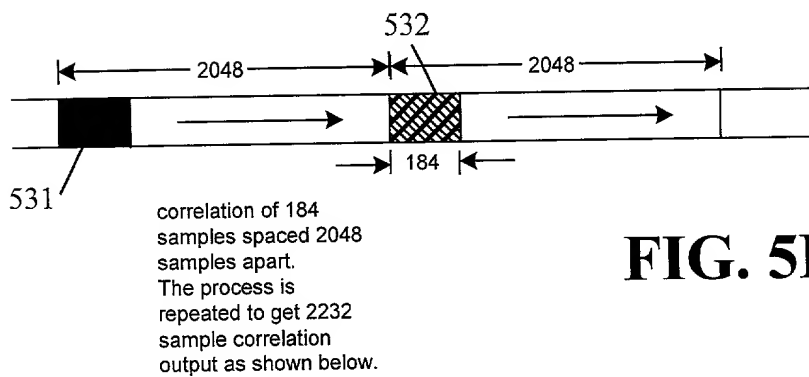


FIG. 5B

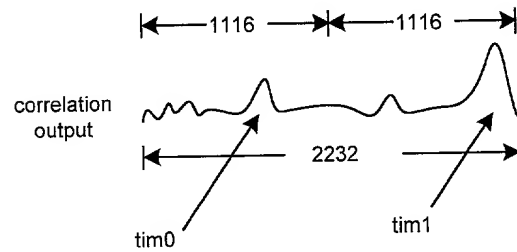


FIG. 5C

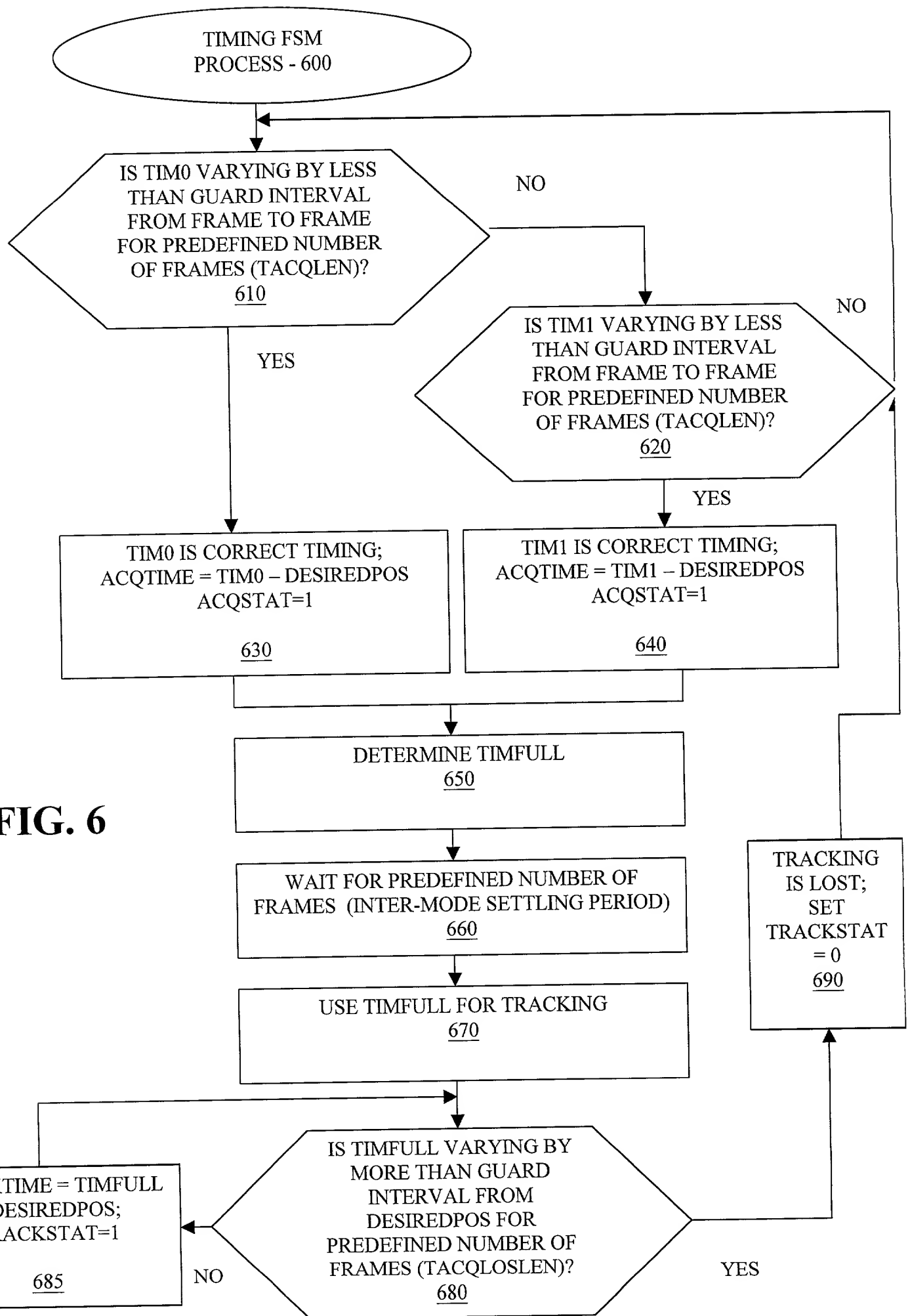


FIG. 6

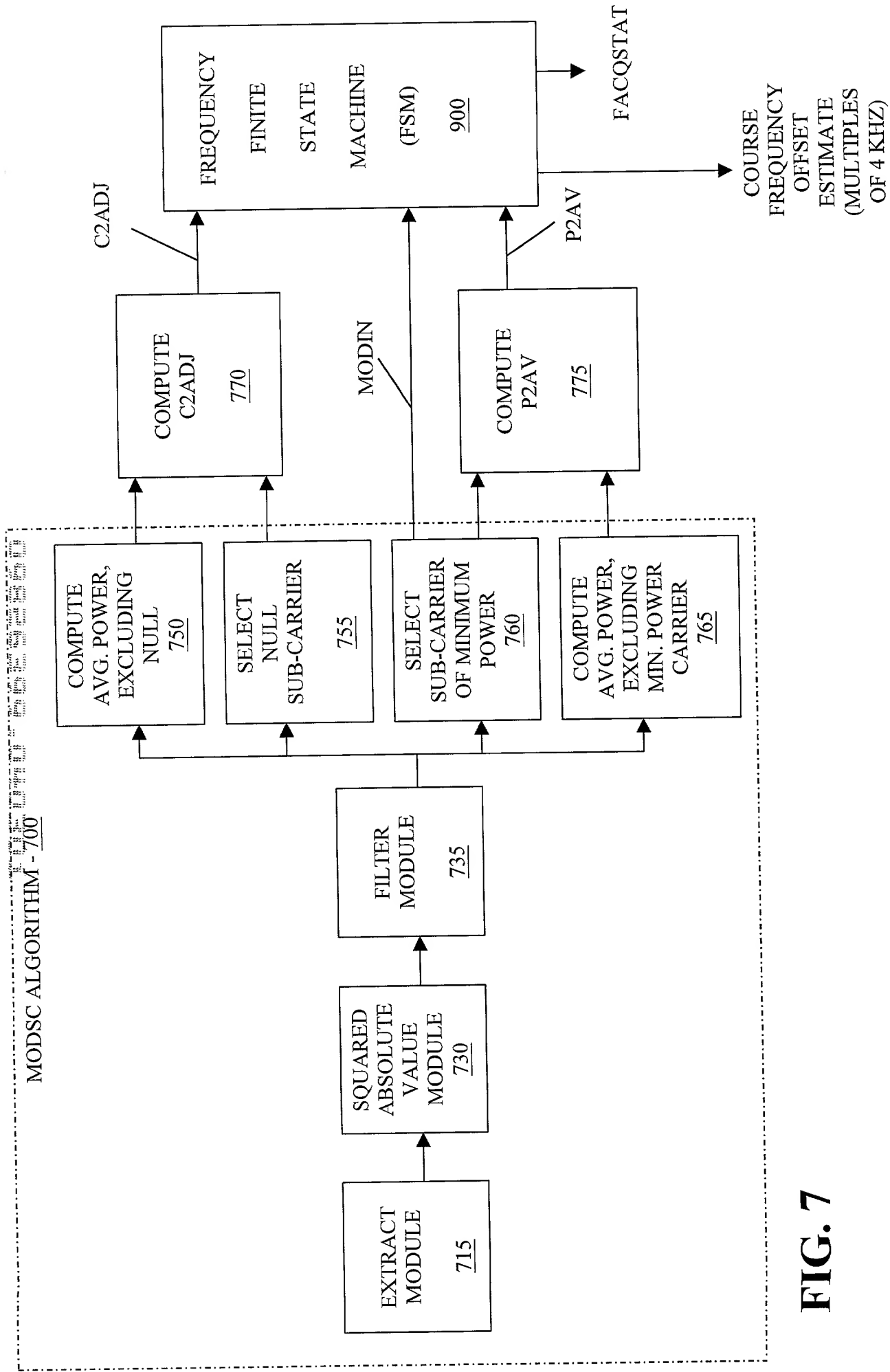


FIG. 7

0	1	18	2030	2047
zero carrier	positive carriers	unused carriers for MODSC		negative carriers

FIG. 8

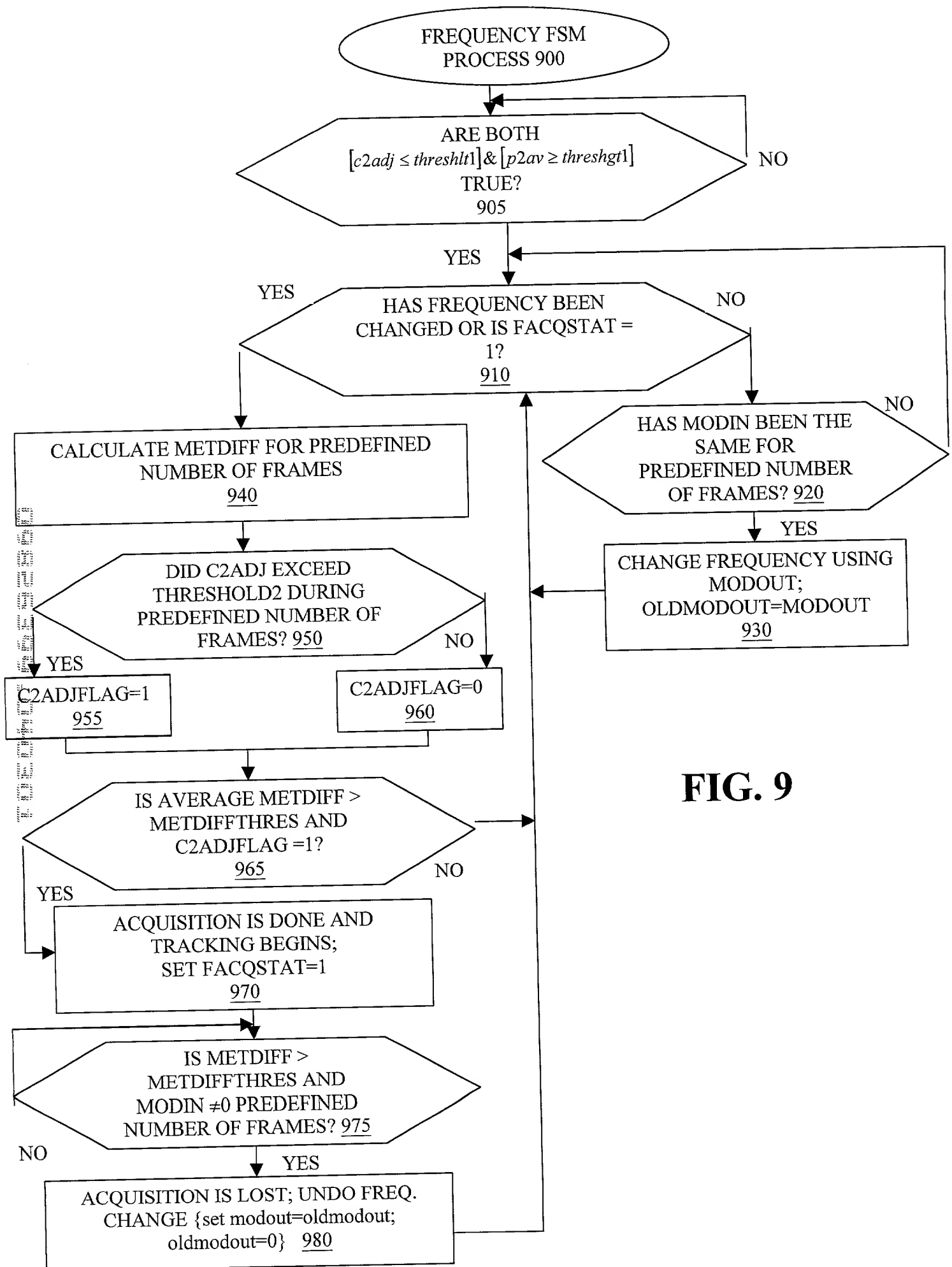


FIG. 9

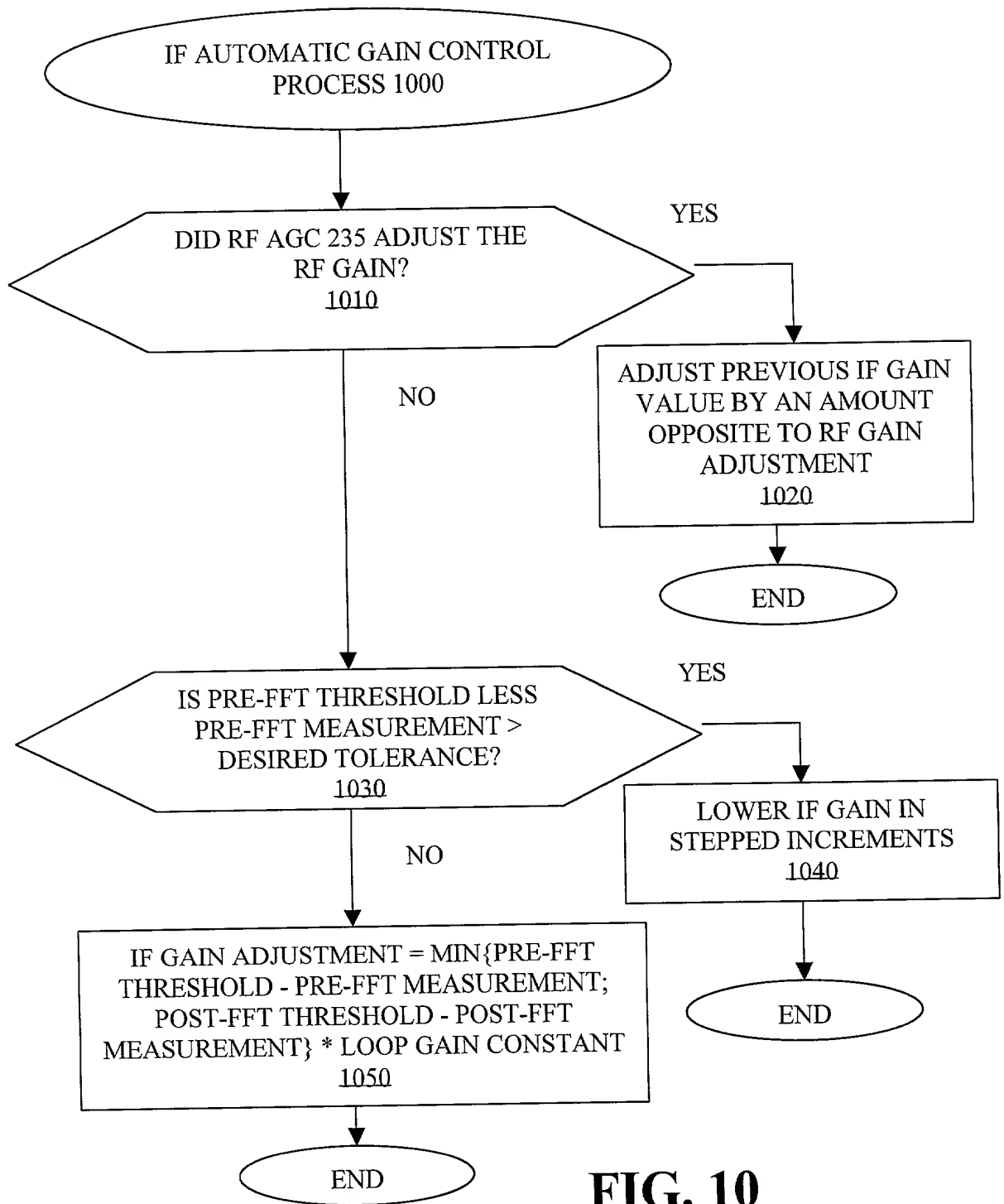


FIG. 10

```

INPUT_PORT(1) register float *Prepower;
INPUT_PORT(2) register float *Postpower;
INPUT_PORT(3) register float *Rfgain;
OUTPUT_PORT(1) register float *Output; /*IF AGC Gain in dB*/

```

```

BLOCKFACTOR long BlockFactor;

```

```

PARAMETER(1) float OutputIntervalWidth; /* 71 dB*/
PARAMETER(2) float SetPointdBPre; /*42.2*/
PARAMETER(3) float SetPointdBPost; /*32.2*/
PARAMETER(4) float Kagc; /*0.25*/
PARAMETER(5) float PreDropdB; /* 3.0*/
PARAMETER(6) long WaitTime; /*8 OFDM Frames!!*/

```

```

STATE float oldoutput;
STATE float oldrfgain;
STATE long counter;

```

```

#include <math.h>

```

```

void init ofdmagecontrol2()
{
/* initialize Sum */
oldoutput = 0.0;
counter = WaitTime;
}

```

```

void ofdmagecontrol2()
{
register float dbinpre, dbinpost, err, rfgain, output;
float HalfInterval = (OutputIntervalWidth / 2.0);

```

FIG. 11A

LOOP(BlockFactor)

```
printf("-----IFbeg-----\n");
dbinpre = *Prepower++; dbinpost = *Postpower++;
rfgain = *RFgain++;

printf("prepower = %f, post = %f, rfgain = %f\n", dbinpre, dbinpost, rfgain);
if((rfgain-oldrfgain)!=0.0)
{
    output = oldoutput -(rfgain-oldrfgain);
    printf("ifgain = -rfdiff = %f, oldrfgain = %f\n", output, oldrfgain);
}
else if ((SetPointdBPre-PreDropdB-dbinpre <=0.0)&& (counter >= WaitTime))
{
    output = oldoutput -(PreDropdB+2.0);
    printf("ifgain = due to Pre = %f\n", -PreDropdB);
    counter=0;
}
else
{
    counter++;
    if(SetPointdBPre-dbinpre < SetPointdBPost-dbinpost)
        err = SetPointdBPre - dbinpre;
    else
        err = SetPointdBPost-dbinpost;
    err = Kage*err;
    output = oldoutput+err;
    printf("output = %f\n", output);
}

if(output>=HalfInterval)
    output = HalfInterval;
else if (output<=-HalfInterval)
    output = -HalfInterval;
else
    output = output;

*Output++ = output;
oldrfgain = rfgain;
oldoutput = output;
printf("-----IFend-----\n");
ENDLOOP
}
```

FIG. 11B